UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/804,852	03/19/2004	Lauri Paatero	915-008.022	7439

4955 7590 02/26/2010

WARE FRESSOLA VAN DER SLUYS & ADOLPHSON, LLP
BRADFORD GREEN, BUILDING 5
755 MAIN STREET, P O BOX 224
MONROE, CT 06468

EXAMINER	
MYLES, YOUNFAA O	

ART UNIT	PAPER NUMBER
2434	

MAIL DATE	DELIVERY MODE
02/26/2010	PAPER

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

Office Action Summary	Application No.	Applicant(s)	
	10/804,852	PAATERO, LAURI	
	Examiner	Art Unit	
	YOUNFAA MYLES	2434	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE *3* MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *22 December 2009*.

2a)☒ This action is **FINAL**. 2b)☐ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1, 4, 6-12* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1,4 and 6-12* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☒ The drawing(s) filed on *09 July 2007* is/are: a)☒ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☐ Information Disclosure Statement(s) (PTO/SB/08) Paper No(s)/Mail Date _____.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____.
5) ☐ Notice of Informal Patent Application
6) ☐ Other: _____.

DETAILED ACTION

[Claim 1-14 status:
There are no new claims,
No claims have been amended,
Claims 1, 4, 6-12 are as previously presented and are pending,
Claims 2, 3, 5, 13 and 14 have been cancelled.]

Examiner's response to (Paatero, 10/804,852) REMARKS

Claims 1 and 4-14 were examined by the Office, and in the Office Action of September 21, 2009 all claims are rejected. With this response claim 5 is cancelled without prejudice. Applicant respectfully requests reconsideration and withdrawal of the rejections in view of the following discussion.

Claim Rejections Under § 112

In section 5, on page 3 of the Office Action, claims 1 and 5 are rejected under 35 U.S.C. § 112, second paragraph, as being indefinite. Applicant respectfully submits that the rejection to claims 1 and 5 is moot in view of the cancellation of claim 5.

[Examiner's response: rejections are so withdrawn as issue of number of physical interfaces is established by claim 1 (without conflict from that otherwise stated in claim 5 – which is now cancelled.]

Claim Rejections Under § 103

In section 6, on page 4 of the Office Action, claims 1, 4-14 are rejected under 35 U.S.C. § 103(a) as unpatentable over Grohoski (U.S. Appl. Publ. No. 2004/0225885) in view of Srinivasan et al. (U.S. Appl. Publ. No. 2004/0158742). Applicant respectfully submits that the cited references, alone or in combination, fail to disclose or suggest all of the limitations recited in claim 1. Claim 1 is amended to recites that the first logical interface is not accessible when data is being transferred in the second logical interface, and the cited references at least fail to disclose or suggest this limitation of claim 1.

[Examiner's response: @Grohoski p[0102] crypto co-processor may be directly accessed via interface ([first, second, etc. of] set of registers) which allow traps, @Grohoski p[0106] control register may control access (to crypto co-processor), enabled bit controls access between processes [and interfaces]; if enabled is not set, such causes [accessibility] trap, @Grohoski p[0176] system may selectively perform [restrictive/alternative data transfer] operation configured by written programs.]

Furthermore, applicant respectfully submits that the cited references also fail to disclose or suggest that the configuration register is configured to receive mode setting instructions from a

protected application.

[Examiner's response: @Grohoski p[0176] system [mode] may be selectively configured in accordance with required purposes, @Grohoski p[0016] control word may include instructions, @Grohoski p[0112, 0113] accordingly, system may have formatted control register with operation field (codes being similar to control words) [/mode setting instructions] which may depict operation(s) to be performed, … an operation is started by writing to [configuration supported] control register, crypto co-processor may arbitrate among control word queue operations, control register fields include … cipher operations where [protected application source] direction field may specify whether encryption (or decryption) is performed, [otherwise] @Grohoski p[0056] crypto co-processor allows crypto [protected application] processing (that may overlap with execution of normal (i.e., non-cryptographic)) instructions. Moreover, @Grohoski p[0144] application [protective] protocol may be set in place before any data is transmitted or received.]

In contrast to claim 1, in Grohoski the crypto co-processor (250) can be accessed through a set of hardware registers, and the crypto co-processor (250) can share memory access units with the main CPU in order to reduce duplicated hardware. See Grohoski paragraph [0056]. The Office appears to assert on page 3 of the Office Action that the common memory corresponds to the first and second logical interfaces as recited in claim 1. In Grohoski the CPU (205) identifies a packet as a crypto packet, then identifies any additional data required to execute the crypto packet, then identifies the additional data in the control queue, and then transfers the crypto packet to the crypto co-processor (250). See Grohoski paragraphs [0058]- [0061]. At time Tl the crypto co-processor (250) receives the crypto packet and retrieves the corresponding control word, and at time T2 the crypto co-processor updates the control word to identify the crypto packet as being completed. See Grohoski paragraph [0062]. However, also between time T1 and time T2 if a subsequent packet is processed in the CPU (205) and identified as a crypto packet, the crypto packet is forwarded to the crypto processor (250) and a corresponding control word is forwarded to the control queue. See Grohoski paragraph [0063]. Accordingly, whenever the crypto co-processor (250) is able to access the memory and/or interface, the CPU (205) is also able to access the same. Therefore, Grohoski fails to disclose or suggest that the first logical interface is not accessible when data is being transferred in the second logical interface, as recited in claim 1.

The Office asserts on page 2 of the Office Action that the applicant has not provided any teaching within Grohoski that supports applicant's argument that both the CPU and the cryptoprocessor can access the memory at the same time in Grohoski. However, the paragraph relied upon by the Office to show that Grohoski teaches that the first logical interface is not accessible when data is transferred in the second logical interface specifically states that the crypto coprocessor (250) also allows crypto processing to overlap with execution of normal (i.e. noncryptographic) instructions. See Grohoski paragraph [0056]. Applicant respectfully submits that at least this discussion in Grohoski makes it apparent that both the CPU and crypto processor can access the memory at the same time. The Office also asserts that since the transfers are made through a shared memory then only one can access it at a time. However, applicant respectfully

disagrees, since some memory such as a multi-ported memory provides more than one access path to its content, and allows the same bank of memory to be read and written simultaneously. See e.g. definition from www.yourdictionary.corn/computer/multiported- memory (attached as Appendix A). However, claim 1 specifically recites that the first logical interface is not accessible when data is transferred in the second logical interface. Therefore for at least the reasons stated above, applicant respectfully submits that Grohoski does not disclose or suggest this limitation of claim 1.

[Examiner's response: argument for multi-ported memory is accepted. Still, positively occurring operation(s) that "can" occur (for a system embodiment) may or may not negate other operations from not formally so occurring; @Kohn p[0031] (as by said @Grohoski p[0057] –incorporated by reference) cryptographic co-processor may send an interrupt [suspension] to an instruction execution unit that is included in the processing core, or where @Kohn +p[0056] instruction may be (non-load/non-store) [memory (in)accessibility] instruction, ... to gain a resultant[/limited] operation where @Grohoski p[0170] second[/first] [interface] operation may be performed [exclusively] separate from first[/second] [interface] operation. Therefore, Applicants' acclaim/argument for XOR-like selectivity of communication interface channel(s) is not appreciated as inventively new or as an improvement thereof.]

Furthermore, on page 5 of the Office Action, the Office acknowledges that Grohoski fails to disclose a configuration register configured to receive mode setting instructions from a protected application, and relies upon Srinivasan for this teaching. However, Srinivasan also fails to disclose or suggest that the configuration register is configured to receive mode setting instructions from a protected application, as recited in claim 1. In contrast to claim 1, Srinivasan only discloses that in a step (216) the trusted server optionally verifies that the secure processor (110) is authorized to receive application software from the trusted server. See Srinivasan paragraph [0105]. However, Srinivasan further states that the CPU operating in secure mode receives the application software or other additional instructions from the trusted server. See Srinivasan paragraph [0107]. If the CPU is already operating in a secure mode before the application software is received from the trusted server, then the application software cannot be considered to be a protected application that provides mode setting instructions to a configuration register, as recited in claim 1.

[Examiner's response: presence of configuration supported control register is noted above; additionally, above argument that "application software cannot be considered to be a protected application ..." does not sufficiently carry logical reason to restrictively establish a "cannot" conclusion. @Srinivansan p[0022] secure processor may enforce [security-related] prevention and protection of application software content wherein such content [instructions] may include set [modes] of permissive [protected] content, digital [selective] execution/presentation rights to enable (application) execution, presentation, or information access.

Moreover, @Srinivansan p[0079] secure processor may have (other) set [modes] of code signatures [instructions] when received from authenticated trusted [protected application] sources.]

In contrast to the present application, in Srinivasan applications corresponding to the protected applications recited in claim 1 are defined as "secure code" and "secure boot loader code." See Srinivasan paragraph [0036]. These protected applications are not the equivalent to the "application soft-ware," which the Office asserts corresponds to the protected applications recited in claim 1. Srinivasan defines "application software" as a set of instructions or parameters capable of being executed or interpreted by a processor. See Srinivasan paragraph [0031]. Since both secure code and application software are defined in the Lexicography provided in Srinivasan, it implies that they are differentiated from each other. Srinivasan makes no mention that the application software is a protected application as mentioned in claim 1.

[Examiner's response: additional to above responses, @Srinivansan claim 35, instructions may include application software, and/or secure code, [where relationally, software and code are operatively/cooperatively associated as by ...] @Srinivansan p[0068] secure boot code may locate and load any software and security functions included in secure code; moreover, lexicography for the phrase "application software" may be considered to cover even "code" as by @Srinivansan p[0031] phrase "application software" describes a set [modes] of instructions or parameters capable of being executed or interpreted by a processor ... concept of application software is broad and is intended to include at least commands or requests to be received and acted upon by an application program, (or any reasonable [code] generalization thereof), and the like. Therefore, arguments for non-equivalency and differentiation are addressed; --application software may be adjudged by given secure context as ascribing to protected application term.]

Therefore, the section relied upon by the Office does not disclose a configuration register configured to receive mode setting instructions from a protected application, as recited in claim 1. Instead, these sections only disclose that the application software places parameters for a request for services in a set of selected registers, or performs an uncached read to a register. See Srinivasan paragraphs [0121] & [0127]. Even if the application software are considered to be a protected application, which applicant does not admit, the functions performed by the application software in Srinivasan do not correspond to providing mode setting instructions, as recited in claim 1.

[Examiner's response: citations and reasons provided above directly address claim 1 concern.]

Furthermore, while Srinivasan defines "secure code" and "secure boot loader code" to be interpretable or executable by the secure processor, and known to the secure processor to be trustable, the secure code and secure boot loader code do not provide mode setting instructions to a configuration register. Claim 1 recites that the configuration register is configured to receive mode setting instructions from a protected application, however even if the secure code and

secure boot loader code are considered to correspond to the protected application, Srinivasan does not disclose a configuration register configured to receive mode setting instructions from the secure code or the secure boot loader code. Instead, after power on of the secure processor (110) a reset signal (A170) is asserted that indicates that the secure processor (110) has been reset. See Srinivasan paragraph [0088]. As a result, the secure mode active signal (A160) is asserted and the CPU transfers execution control to the secure boot code (A115). The secure mode active signal (A160) indicates to the non-volatile memory that the CPU is allowed to access the secure boot code, execute its instruction, and read and write data using the security information (113). See Srinivasan paragraph [0089]. However, Srinivasan does not disclose or suggest that a configuration register receives mode setting instructions from a protected application, instead it appears that the reset signal (A170) is responsible for setting the secure processor (110). Therefore, for at least these reasons claim 1 is not disclosed or suggested by the cited references.

[Examiner's response: citations and reasons provided above directly address claim 1 concern. Positive teachings of references are provided to map to applications' teachings.]

Independent claim 12 is amended in a manner similar to claim 1, and contains limitations similar to claim 1. Therefore, for at least the reasons discussed above in relation to claim 1, claim 12 is not disclosed or suggested by the cited references.

The dependent claims depending from the above mentioned independent claims are not disclosed or suggested by the cited references at least in view of their dependencies.

[Examiner's response: argument for claims 1 and 12 are adjudged similarly as non-persuasive (by standard of preponderance) in overcoming claim rejections. Standing dependent claims are accorded likewise stance due to previously asserted rejection reasons (which have not been herein specifically traversed) and by independent claim association.]

Conclusion to Examiner's Response

It is therefore respectfully submitted that the present application is in condition for allowance and such action is earnestly solicited.

[Examiner's response: terms have been mapped, and remarks/amendments have been addressed –wherein focus of analysis has been placed on utility functionality as opposed to administrative/design layout(s).]

DETAILED ACTION (Continued)

[Restated claim (1-14) status:
There are no new claims,
No claims have been amended,
Claims 1, 4, 6-12 are as previously presented and are pending,
Claims 2, 3, 5, 13 and 14 have been cancelled.]

1. Claims 1, 4, 6-12 are [were] pending.

Response to Arguments

2. Applicant's arguments filed 7/15/2009 have been fully considered but they are not persuasive.

3. Applicant has argued that Grohoski fails to teach the first logical interface is not accessible when data is transferred in the second logical interface. Examiner respectfully disagrees. Applicant has asserted that both the CPU and the cryptoprocessor can access the memory at the same time. However, Applicant has not pointed to any teaching within the reference that supports such an assertion. Applicant has cited portions of the reference referring to time T1, time T2, and some time between T1 and T2, but the CPU or cryptoprocessor does not access the first interface while the second interface is in use at any of these cited times. The Grohoski teaches the first logical interface is not accessible when data is transferred in the second logical interface (Grohoski, paragraph 0056, paragraphs 0061-0062). Grohoski teaches the limitation by teaching a CPU transferring data through a first and second interface. The

limitation in question requires that the first logical interface (data transfer interface) not be accessible when the second logical interface (key transfer interface) is in use for transfer. Transfers are made through a set of shared memory (Grohoski, paragraph 0056). Accordingly, nothing can be read from the memory while a write operation is taking place. A computer memory cannot be read while it is being written to. As a result, Examiner maintains that Grohoski teaches the limitation in question.

4. Applicant further argues that Srinivasan fails to teach a "protected application." Examiner respectfully disagrees. Srinivasan teaches a configuration register arranged to indicate to the accelerator whether secure mode or normal mode is set by the processor and configured to receive mode setting instructions from a protected application, said processor arranged in the device (Srinivasan, paragraphs 0121-0123, 0127, 0133, paragraph 0139). Srinivasan teaches the application is protected by teaching application software setting parameters in registers (Srinivasan, paragraph 0121) which causes secure mode to be entered (Srinivasan, paragraphs 0121-0123, paragraphs 0133, 0139). The secure mode may then perform secure operations for the application including verifying additional components of the protected application (Srinivasan, paragraphs 0133, 0139).

Claim Rejections - 35 USC § 103

The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person

having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

5. **Claims 1, 4, 6-12 are rejected under 35 U.S.C. 103(a)** as being unpatentable over Grohoski et al US PGPub 2004/0225885 in view of Srinivasan et al US PGPub 2004/0158742.

6. **With regards to claims 1, 11-12**, Grohoski teaches an electronic device comprising (Grohoski, paragraph 0056, paragraph 0106, crypto processor), an accelerator for accelerating cryptographic data processing operations, which acceleration is arranged with (Grohoski, paragraph 0056, higher speed encryption and decryption processes enabled using crypto coprocessor) a first logical interface over which data to be processed is provided (Grohoski, paragraphs 0061-0062, transfers crypto packet), a secure second logical interface over which cryptographic keys employed in the operation of processing data is provided (Grohoski, paragraph 0062, paragraph 0052, control queue, paragraphs 0056-0057, sharing access to registers and memory access units provides a secure connection, paragraph 0106, controlled access to secure registers), and wherein the first logical interface and the secure second logical interface share a same physical interface (Grohoski, paragraph 0056, share same memory access units) and wherein the first logical interface is not accessible when data is transferred in the second logical interface (Grohoski, paragraph 0056, paragraphs 0061-0062). Grohoski fails to teach a configuration register arranged to indicate to the accelerator whether secure mode or normal mode is set by the processor arranged in the device. However, Srinivasan teaches a configuration register arranged to indicate to the accelerator whether secure mode or normal mode is set by the processor and configured to receive mode setting instructions from a protected application, said processor arranged in the device (Srinivasan, paragraphs 0121, 0127, 0133). At the time the invention was made, it would have been obvious to a person of ordinary skill in the

art to utilize Srinivasan's method of providing secure operation modes for a processor because it offers the advantage of ensuring that only authorized application software is executed and only authorized multimedia content is rendered (Srinivasan, paragraph 0007).

7. **With regards to claim 4**, Grohoski as modified teaches the configuration register is further arranged such that is may be set in one of a plurality of possible encryption modes, the accelerator being arranged to operate in the encryption mode set in the register (Grohoski, paragraph 0116, encryption type field).

8. **With regards to claim 6**, Grohoski as modified teaches the first logical interface of the accelerator is arranged such that it is accessible by any application while the secure second logical interface of the accelerator is arranged such that it is accessible by protected applications only (Srinivasan, paragraphs 0007, 0121, 0127, 0133).

9. **With regards to claim 7**, Grohoski as modified teaches protected applications prevent other applications from accessing the accelerator (Grohoski, paragraph 0106).

10. **With regards to claim 8**, Grohoski as modified teaches protected applications are applications which are allowed to execute in the secure execution environment (Srinivasan, paragraphs 0121, 0127, 0133, Abstract).

11. **With regards to claim 9**, Grohoski as modified teaches storage circuitry arranged with at least one storage area in which protected data relating to device security is located (Grohoski, paragraph 0106), a processor arranged such that is may be set in one of at least two separate operating modes (Srinivasan, paragraphs 0007, 0121, 0127, 0133) and the device further arranged such that the processor is given access to said storage area when a normal processor operating mode is set (Srinivasan, paragraphs 0007, 0121, 0127, 0133) and the processor is

denied access to said storage area when a normal processor operating mode is set (Srinivasan, paragraphs 0007, 0121, 0127, 0133) and the processor is capable of accessing the secure second logical interface of the accelerator when the secure processor operating mode is set (Srinivasan, paragraphs 0007, 0121, 0127, 0133).

12. **With regards to claim 10**, Grohoski as modified teaches the protected applications controlling the processor operation mode (Srinivasan, paragraph 0010).

Conclusion

Applicant's amendment necessitated no new ground(s) of rejection for this Office action. Accordingly, **THIS ACTION IS MADE FINAL**. See MPEP § 706.07(a). Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire THREE MONTHS from the mailing date of this action. In the event a first reply is filed within TWO MONTHS of the mailing date of this final action and the advisory action is not mailed until after the end of the THREE-MONTH shortened statutory period, then the shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of the advisory action. In no event, however, will the statutory period for reply expire later than SIX MONTHS from the date of this final action.

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Youn Myles whose telephone number is (571) 270-3358. The examiner can normally be reached 8:00-5:30 Mon-Fri.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Kambiz Zand can be reached on (571) 272-3811. The fax phone number for the organization where this application or proceeding is assigned is 571-273-3811.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications

may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

/YOUNFAA MYLES/

Examiner, Art Unit 2434

2/16/2009

/Kambiz Zand/

Supervisory Patent Examiner, Art Unit 2434